Network Working Group
Request for Comments: 5625
BCP: 152
Category: Best Current Practice

R. Bellis
Nominet UK

August 2009

DNS Proxy Implementation Guidelines

Abstract

This document provides guidelines for the implementation of DNS proxies, as found in broadband gateways and other similar network devices.

Status of This Memo

This document specifies an Internet Best Current Practices for the Internet Community, and requests discussion and suggestions for improvements. Distribution of this memo is unlimited.

Table of Contents

1. Introduction

Research has found ([SAC035], [DOTSE]) that many commonly used broadband gateways (and similar devices) contain DNS proxies that are incompatible in various ways with current DNS standards.

These proxies are usually simple DNS forwarders, but typically do not have any caching capabilities. The proxy serves as a convenient default DNS resolver for clients on the LAN, but relies on an upstream resolver (e.g., at an ISP) to perform recursive DNS lookups.

Note that to ensure full DNS protocol interoperability it is preferred that client stub resolvers should communicate directly with full-feature, upstream recursive resolvers wherever possible.

That notwithstanding, this document describes the incompatibilities that have been discovered and offers guidelines to implementors on how to provide better interoperability in those cases where the client must use the broadband gateway's DNS proxy.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. The Transparency Principle

It is not considered practical for a simple DNS proxy to implement all current and future DNS features.

There are several reasons why this is the case:

- Broadband gateways usually have limited hardware resources.
- Firmware upgrade cycles are long, and many users do not routinely apply upgrades when they become available.
- No one knows what those future DNS features will be or how they might be implemented.
- Doing so would substantially complicate the configuration user interface (UI) of the device.

Furthermore, some modern DNS protocol extensions (see, e.g., EDNS0 below) are intended to be used as "hop-by-hop" mechanisms. If the DNS proxy is considered to be such a "hop" in the resolution chain, then for it to function correctly, it would need to be fully compliant with all such mechanisms.

[SAC035] shows that the more actively a proxy participates in the DNS protocol, the more likely it is that it will somehow interfere with the flow of messages between the DNS client and the upstream recursive resolvers.

The role of the proxy should therefore be no more and no less than to receive DNS requests from clients on the LAN side, forward those verbatim to one of the known upstream recursive resolvers on the WAN side, and ensure that the whole response is returned verbatim to the original client.

It is RECOMMENDED that proxies should be as transparent as possible, such that any "hop-by-hop" mechanisms or newly introduced protocol extensions operate as if the proxy were not there.

Except when required to enforce an active security or network policy (such as maintaining a pre-authentication "walled garden"), end-users SHOULD be able to send their DNS queries to specified upstream

resolvers, thereby bypassing the proxy altogether. In this case, the gateway SHOULD NOT modify the DNS request or response packets in any way.

4. Protocol Conformance

4.1. Unexpected Flags and Data

The Transparency Principle above, when combined with Postel's Robustness Principle [RFC0793], suggests that DNS proxies should not arbitrarily reject or otherwise drop requests or responses based on perceived non-compliance with standards.

For example, some proxies have been observed to drop any packet containing either the "Authentic Data" (AD) or "Checking Disabled" (CD) bits from DNSSEC [RFC4035]. This may be because [RFC1035] originally specified that these unused "Z" flag bits "MUST" be zero. However, these flag bits were always intended to be reserved for future use, so refusing to proxy any packet containing these flags (now that uses for those flags have indeed been defined) is not appropriate.

Therefore, proxies MUST ignore any unknown DNS flags and proxy those packets as usual.

4.2. Label Compression

Compression of labels as per Section 4.1.4 of [RFC1035] is optional.

Proxies MUST forward packets regardless of the presence or absence of compressed labels therein.

4.3. Unknown Resource Record Types

[RFC3597] requires that resolvers MUST handle Resource Records (RRs) of unknown type transparently.

All requests and responses MUST be proxied regardless of the values of the QTYPE and QCLASS fields.

Similarly, all responses MUST be proxied regardless of the values of the TYPE and CLASS fields of any Resource Record therein.

4.4. Packet Size Limits

[RFC1035] specifies that the maximum size of the DNS payload in a UDP packet is 512 octets. Where the required portions of a response would not fit inside that limit, the DNS server MUST set the

"TrunCation" (TC) bit in the DNS response header to indicate that truncation has occurred. There are however two standard mechanisms (described in Sections 4.4.1 and 4.4.2) for transporting responses larger than 512 octets.

Many proxies have been observed to truncate all responses at 512 octets, and others at a packet size related to the WAN MTU, in either case doing so without correctly setting the TC bit.

Other proxies have been observed to remove the TC bit in server responses that correctly had the TC bit set by the server.

If a DNS response is truncated but the TC bit is not set, then client failures may result. In particular, a naive DNS client library might suffer crashes due to reading beyond the end of the data actually received.

Since UDP packets larger than 512 octets are now expected in normal operation, proxies SHOULD NOT truncate UDP packets that exceed that size. See Section 4.4.3 for recommendations for packet sizes exceeding the WAN MTU.

If a proxy must unilaterally truncate a response, then the proxy MUST set the TC bit. Similarly, proxies MUST NOT remove the TC bit from responses.

4.4.1. TCP Transport

Should a UDP query fail because of truncation, the standard fail-over mechanism is to retry the query using TCP, as described in Section 6.1.3.2 of [RFC1123].

Whilst TCP transport is not strictly mandatory, it is supported by the vast majority of stub resolvers and recursive servers. Lack of support in the proxy prevents this fail-over mechanism from working.

DNS proxies MUST therefore be prepared to receive and forward queries over TCP.

Note that it is unlikely that a client would send a request over TCP unless it had already received a truncated UDP response. Some "smart" proxies have been observed to first forward any request received over TCP to an upstream resolver over UDP, only for the response to be truncated, causing the proxy to retry over TCP. Such behaviour increases network traffic and causes delay in DNS resolution since the initial UDP request is doomed to fail.

Therefore, whenever a proxy receives a request over TCP, the proxy SHOULD forward the query over TCP and SHOULD NOT attempt the same query over UDP first.

4.4.2. Extension Mechanisms for DNS (EDNS0)

The "Extension Mechanism for DNS" [RFC2671] was introduced to allow the transport of larger DNS packets over UDP and also to allow for additional request and response flags.

A client may send an OPT Resource Record (OPT RR) in the Additional Section of a request to indicate that it supports a specific receive buffer size. The OPT RR also includes the "DNSSEC OK" (DO) flag used by DNSSEC to indicate that DNSSEC-related RRs should be returned to the client.

However, some proxies have been observed to either reject (with a FORMERR response code) or black-hole any packet containing an OPT RR. As per Section 4.1, proxies MUST NOT refuse to proxy such packets.

4.4.3. IP Fragmentation

Support for UDP packet sizes exceeding the WAN MTU depends on the gateway's algorithm for handling fragmented IP packets. Several methods are possible:

1. Fragments are dropped.

2. Fragments are forwarded individually as they're received.

3. Complete packets are reassembled on the gateway and then re-fragmented (if necessary) as they're forwarded to the client.

Method 1 above will cause compatibility problems with EDNS0 unless the DNS client is configured to advertise an EDNS0 buffer size limited to the WAN MTU less the size of the IP header. Note that RFC 2671 does recommend that the path MTU should be taken into account when using EDNS0.

Also, whilst the EDNS0 specification allows for a buffer size of up to 65535 octets, most common DNS server implementations do not support a buffer size above 4096 octets.

Therefore (irrespective of which of the above methods is in use), proxies SHOULD be capable of forwarding UDP packets up to a payload size of at least 4096 octets.

NB: in theory, IP fragmentation may also occur if the LAN MTU is smaller than the WAN MTU, although the author has not observed such a configuration in use on any residential broadband service.

4.5. Secret Key Transaction Authentication for DNS (TSIG)

[RFC2845] defines TSIG, which is a mechanism for authenticating DNS requests and responses at the packet level.

Any modifications made to the DNS portions of a TSIG-signed query or response packet (with the exception of the Query ID) will cause a TSIG authentication failure.

DNS proxies MUST implement Section 4.7 of [RFC2845] and either forward packets unchanged (as recommended above) or fully implement TSIG.

As per Section 4.3, DNS proxies MUST be capable of proxying packets containing TKEY [RFC2930] Resource Records.

NB: any DNS proxy (such as those commonly found in WiFi hotspot "walled gardens") that transparently intercepts all DNS queries and that returns unsigned responses to signed queries, will also cause TSIG authentication failures.

5. DHCP's Interaction with DNS

Whilst this document is primarily about DNS proxies, most consumers rely on DHCP [RFC2131] to obtain network configuration settings. Such settings include the client machine's IP address, subnet mask, and default gateway, but also include DNS-related settings.

It is therefore appropriate to examine how DHCP affects client DNS configuration.

5.1. Domain Name Server (DHCP Option 6)

Most gateways default to supplying their own IP address in the DHCP "Domain Name Server" option [RFC2132]. The net result is that without explicit re-configuration many DNS clients will, by default, send queries to the gateway's DNS proxy. This is understandable behaviour given that the correct upstream settings are not usually known at boot time.

Most gateways learn their own DNS settings via values supplied by an ISP via DHCP or PPP over the WAN interface. However, whilst many gateways do allow the device administrator to override those values, some gateways only use those supplied values to affect the proxy's own forwarding function, and do not offer these values via DHCP.

When using such a device, the only way to avoid using the DNS proxy is to hard-code the required values in the client operating system. This may be acceptable for a desktop system but it is inappropriate for mobile devices that are regularly used on many different networks.

As per Section 3, end-users SHOULD be able to send their DNS queries directly to specified upstream resolvers, ideally without hard-coding those settings in their stub resolver.

It is therefore RECOMMENDED that gateways SHOULD support device-administrator configuration of values for the "Domain Name Server" DHCP option.

5.2. Domain Name (DHCP Option 15)

A significant amount of traffic to the DNS Root Name Servers is for invalid top-level domain names, and some of that traffic can be attributed to particular equipment vendors whose firmware defaults this DHCP option to specific values.

Since no standard exists for a "local" scoped domain name suffix, it is RECOMMENDED that the default value for this option SHOULD be empty, and that this option MUST NOT be sent to clients when no value is configured.

5.3. DHCP Leases

It is noted that some DHCP servers in broadband gateways offer, by default, their own IP address for the "Domain Name Server" option (as described above) but then automatically start offering the upstream servers' addresses once they've been learnt over the WAN interface.

In general, this behaviour is highly desirable, but the effect for the end-user is that the settings used depend on whether the DHCP lease was obtained before or after the WAN link was established.

If the DHCP lease is obtained whilst the WAN link is down, then the DHCP client (and hence the DNS client) will not receive the correct values until the DHCP lease is renewed.

Whilst no specific recommendations are given here, vendors may wish to give consideration to the length of DHCP leases and to whether some mechanism for forcing a DHCP lease renewal might be appropriate.

Another possibility is that the learnt upstream values might be persisted in non-volatile memory such that on reboot the same values can be automatically offered via DHCP. However, this does run the risk that incorrect values are initially offered if the device is moved or connected to another ISP.

Alternatively, the DHCP server might only issue very short (i.e., 60 second) leases while the WAN link is down, only reverting to more typical lease lengths once the WAN link is up and the upstream DNS servers are known. Indeed, with such a configuration it may be possible to avoid the need to implement a DNS proxy function in the broadband gateway at all.

6. Security Considerations

This document introduces no new protocols. However, there are some security-related recommendations for vendors that are listed here.

6.1. Forgery Resilience

Whilst DNS proxies are not usually full-feature resolvers, they nevertheless share some characteristics with them.

Notwithstanding the recommendations above about transparency, many DNS proxies are observed to pick a new Query ID for outbound requests to ensure that responses are directed to the correct client.

NB: changing the Query ID is acceptable and compatible with proxying TSIG-signed packets since the TSIG signature calculation is based on the original message ID, which is carried in the TSIG RR.

It has been standard guidance for many years that each DNS query should use a randomly generated Query ID. However, many proxies have been observed picking sequential Query IDs for successive requests.

It is strongly RECOMMENDED that DNS proxies follow the relevant recommendations in [RFC5452], particularly those in Section 9.2 relating to randomisation of Query IDs and source ports. This also applies to source port selection within any NAT function.

If a DNS proxy is running on a broadband gateway with NAT that is compliant with [RFC4787], then it SHOULD also follow the recommendations in Section 10 of [RFC5452] concerning how long DNS state is kept.

6.2. Interface Binding

Some gateways have been observed to have their DNS proxy listening on both internal (LAN) and external (WAN) interfaces. In this configuration, it is possible for the proxy to be used to mount reflector attacks as described in [RFC5358].

The DNS proxy in a gateway SHOULD NOT, by default, be accessible from the WAN interfaces of the device.

6.3. Packet Filtering

The Transparency and Robustness Principles are not entirely compatible with the deep packet-inspection features of security appliances such as firewalls, which are intended to protect systems on the inside of a network from rogue traffic.

However, a clear distinction may be made between traffic that is intrinsically malformed and that which merely contains unexpected data.

Examples of malformed packets that MAY be dropped include:

- invalid compression pointers (i.e., those that point outside of the current packet or that might cause a parsing loop)
- incorrect counts for the Question, Answer, Authority, and Additional Sections (although care should be taken where truncation is a possibility)

Dropped packets will cause the client to repeatedly retransmit the original request, with the client only detecting the error after several retransmit intervals.

In these circumstances, proxies SHOULD synthesise a suitable DNS error response to the client (i.e., SERVFAIL) instead of dropping the packet completely. This will allow the client to detect the error immediately.

7. Acknowledgements

The author would particularly like to acknowledge the assistance of Lisa Phifer of Core Competence. In addition, the author is grateful for the feedback from the members of the DNSEXT Working Group.

8. References

8.1. Normative References

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[RFC1123] Braden, R., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, October 1989.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, March 1997.

[RFC2671] Vixie, P., "Extension Mechanisms for DNS (EDNS0)", RFC 2671, August 1999.

[RFC2845] Vixie, P., Gudmundsson, O., Eastlake, D., and B. Wellington, "Secret Key Transaction Authentication for DNS (TSIG)", RFC 2845, May 2000.

[RFC2930] Eastlake, D., "Secret Key Establishment for DNS (TKEY RR)", RFC 2930, September 2000.

[RFC3597] Gustafsson, A., "Handling of Unknown DNS Resource Record (RR) Types", RFC 3597, September 2003.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, March 2005.

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

[RFC5358] Damas, J. and F. Neves, "Preventing Use of Recursive Nameservers in Reflector Attacks", BCP 140, RFC 5358, October 2008.

[RFC5452] Hubert, A. and R. van Mook, "Measures for Making DNS More Resilient against Forged Answers", RFC 5452, January 2009.

8.2. Informative References

[DOTSE] Ahlund and Wallstrom, "DNSSEC Tests of Consumer Broadband Routers", February 2008, <http://www.iis.se/docs/Routertester_en.pdf>.

[SAC035] Bellis, R. and L. Phifer, "Test Report: DNSSEC Impact on Broadband Routers and Firewalls", September 2008, <http://www.icann.org/committees/security/sac035.pdf>.

Author's Address

Ray Bellis
Nominet UK
Edmund Halley Road
Oxford OX4 4DQ
United Kingdom

Phone: +44 1865 332211
EMail: ray.bellis@nominet.org.uk
URI: http://www.nominet.org.uk/